Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

A Multi-Pass Sieve for Name Normalization

Jennifer D'Souza
Human Language Technology Research Institute
University of Texas at Dallas, Richardson, TX 75083-0688
jennifer.l.dsouza@utdallas.edu

Abstract

We propose a simple multi-pass sieve framework that applies tiers of deterministic normalization modules one at a time from highest to lowest precision for the task of normalizing names. While a sieve based architecture has been shown effective in coreference resolution, it has not yet been applied to the normalization task. We find that even in this task, the approach retains its characteristic features of being simple, and highly modular. In addition, it also proves robust when evaluated on two different kinds of data: clinical notes and biomedical text, by demonstrating high accuracy in normalizing disorder names found in both datasets.

Introduction

Often in natural language, one finds the same concept being referred to with varying names. For example, swelling of abdomen, abdominal swelling, swollen abdomen, abdominal distention, etc., are all synonymous names essentially referring to the same concept. Without a name-to-concept mapping mechanism, such varied forms of naming a concept can prove quite problematic to information retrieval systems (e.g., search engines). Name normalization facilitates precisely such a mapping.

The task of name normalization, as it is defined, involves the mapping of a phrase/word to a unique concept in an ontology (based on the description of that concept in the ontology) after disambiguating potential ambiguous surface words, or phrases, and then assigning the concept's unique identifier in the ontology to the name.

Multi-Pass Sieve Approach

Most existing name normalization systems use tf-idf vector space models (VSM) to represent tokens of an entire thesaurus of concept names; these VSMs are then employed for normalizing new names to concepts. Such approaches tend to be complex in terms of both time and storage. To overcome these complexities, we propose using a simple multi-pass sieve system. This multi-pass sieve framework is organized as tiers of name normalization modules that produce variations of a name via morphological or syntactic transformations one at a time from highest to lowest precision of specificity of the name. Each tier builds on the names constructed by previous sieve modules, guaranteeing that stronger name tranformation methods are given precedence over weaker ones.

In the past, a multi-pass sieve framework was successfully applied to the coreference resolution task (Raghunathan et al. 2010). We show that this framework, albeit with an entirely different set of sieves, is also suited for name normalization in clinical/biomedical data. Below are the sieves in our framework.[1]

Sieve 1 - Exact Match. Names are normalized if they match exactly and unambiguously with existing concept names in the ontology. Names not normalized in this sieve pass through to the next sieve.

Sieve 2 - Abbreviation Expansion. Names are normalized if their expanded version (e.g., *klebsiella pna* expanded to *klebsiella pneumonia*) satisfies the exact match criterion. Abbreviations are expanded based on the algorithm in Schwartz and Hearst (2003) or from the Wikipedia list of medical abbreviations. The remaining names in original and expanded form pass through to the next sieve.

Sieve 3 - Subject ⇔ Object Conversion. Names are normalized if any of their new forms produced in this sieve matches a concept name in the ontology. New forms of a name are obtained by: 1) replacing the preposition in the name with other prepositions (e.g., *changes on ekg* converted to *changes in ekg*); 2) dropping the preposition from the name and swapping the substrings surrounding it (e.g., *changes on ekg* converted to *ekg changes*); 3) bringing the last token to the front, inserting a preposition as the second token, and shifting the remaning tokens to right by two (e.g., *mental status alteration* coverted to *alteration in mental status*); and 4) moving the first token to the end, inserting a preposition as the second to last token, and shifting the remaining tokens to the left by two (e.g., *leg cellulitis* converted to *cellulitis of leg*).

Sieve 4 - Numbers Replacement. For names containing numbers between one to ten, new names are produced by replacing the number in the name with its other forms. We

[1] Additional data used in the normalization system is available at http://www.hlt.utdallas.edu/~jld082000/normalization/

consider the numeral, roman numeral, cardinal, and multiplicative forms of a number for mapping and replacement. For example, *three vessel disease* becomes the set {*3 vessel disease*, *iii vessel disease*, and *triple vessel disease*}. Normalization is by exact match with a concept name.

Sieve 5 - Disorder Synonyms Replacement. For names containing a disorder term, new names are created by replacement of the disorder term with its synonyms.[2] For example, *presyncopal events* becomes {*presyncopal disorders*, *presyncopal episodes*, etc.}. And for names that do not contain a disorder term, new names are created by appending the given name with a disorder synonym. E.g., *crohns* becomes {*crohns disease*, *crohns disorder*, etc.}. Normalization of these new names is by exact match with a concept name in the ontology.

Sieve 6 - Affixation. Names satisfying affixation patterns observed in training data are affixated in this sieve. For example, *infectious source* became *source of infectious* in sieve 3, which then becomes *source of infection* in this sieve. As in earlier sieves, normalization of affixated forms of names is by exact match with a concept name.

Sieve 7 - Stemming. Names are stemmed using the Porter stemmer, and then checked for normalization by exact match with stemmed concept names.

Sieve 8 - Partial Match. Unlike earlier sieves, where new names were created for names produced from all preceding sieves, this sieve module uses only the output from sieve 2 i.e. either the abbreviation expanded form of the name, if available, or the given name itself. In clinical notes, the following criteria were used for normalization by partial match: if the given name tokens matched exactly, irrespective of order, with tokens of a concept name; or for names containing two or more tokens, if a concept name in the ontology matched unambiguously with its tail substring. In biomedical text, names were normalized to the concept with which it shared the most name tokens. And in the case of ties, the concept with the shortest name length was preferred.

Note that, as not all names are included as concepts in the ontology, a subset of the annotated disorder names do not have concept identifiers. Therefore, the default setting of our framework was to normalize names as “CUI-less”.

Corpora

We used the following corpora in our experiments:

Clinical Notes - The ShARe/CLEF eHealth Challenge (Pradhan et al. 2013) corpus contained 199 notes for training and 99 notes for testing. Concept identifiers from training data and from the UMLS Metathesaurus (Campbell, Oliver, and Shortliffe 1998) were used for normalizing names from this corpus.

Biomedical Abstracts - The NCBI disease corpus (Doğan, Leaman, and Lu 2014) contained 693 abstracts for training and development, and 100 abstracts for testing. Concept identifiers from training data and from the MEDIC (Davis et al. 2012) lexicon were used for names from this corpus.

	$\text{Acc}_{ShARecorpus}$	$\text{Acc}_{NCBIcorpus}$
Sieve 1	83.2	69.4
+ Sieve 2	84.5	74.5
+ Sieve 3	84.7	74.6
+ Sieve 4	84.8	75.5
+ Sieve 5	85.2	76.5
+ Sieve 6	85.9	77.1
+ Sieve 7	87.7	78.8
+ Sieve 8	**88.3**	**83.3**

Table 1: Normalization accuracies on test data from the ShARe corpus and the NCBI corpus, respectively, as sieves are added to the multi-pass sieve system.

Results and Conclusion

We present the results of our approach in Table 1. Performance is reported as the percentage accuracy in normalizing names. Our sieve model outperforms the best performance (Leaman, Doğan, and Lu 2013) reported at 82.2 on the NCBI corpus, and ranks close to the best performance (Ghiasvand and Kate 2014) reported at 89.5 on the ShARe corpus. Thus we have shown that in spite of its simplicity, a multi-pass sieve approach is suitable to the normalization task. In addition, its modular architecture is highly advantageous to facilitating further improvement without any major changes to the existing modules.

Acknowledgements

With thanks to the two reviewers for helpful feedback on an earlier version of this article.

References

Campbell, K. E.; Oliver, D. E.; and Shortliffe, E. H. 1998. The unified medical language system: Towards a collaborative approach for solving terminologic problems. *J Am Med Inform Assoc* 5(1):12–16.

Davis, A. P.; Wiegers, T. C.; Rosenstein, M. C.; and Mattingly, C. J. 2012. Medic: a practical disease vocabulary used at the comparative toxicogenomics database. *Database* 2012:bar065.

Doğan, R. I.; Leaman, R.; and Lu, Z. 2014. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *J Biomed inform* 47:1–10.

Ghiasvand, O., and Kate, R. 2014. Uwm: Disorder mention extraction from clinical text using crfs and normalization using learned edit distance patterns. In *Proceedings of SemEval 2014*, 828–832.

Leaman, R.; Doğan, R. I.; and Lu, Z. 2013. Dnorm: disease name normalization with pairwise learning to rank. *Bioinformatics* 2909–2917.

Porter, M. F. 1980. An algorithm for suffix stripping. *Program* 14(3):130–137.

Pradhan, S.; Elhadad, N.; South, B.; Martinez, D.; Christensen, L.; Vogel, A.; Suominen, H.; Chapman, W.; and Savova, G. 2013. Task 1: Share/clef ehealth evaluation lab 2013. *Online Working Notes of CLEF, CLEF* 230.

Raghunathan, K.; Lee, H.; Rangarajan, S.; Chambers, N.; Surdeanu, M.; Jurafsky, D.; and Manning, C. 2010. A multi-pass sieve for coreference resolution. In *Proc. of the 2010 Conference on EMNLP*, 492–501.

Schwartz, A., and Hearst, M. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Proc. of the 8th PSB*, 451–462.

[2] A list of synonyms of the disorder concept are obtained from training data.